IN THE CLAIMS

Please amend the claims as follows:

1. (Previously Presented) An apparatus comprising:

a load/store unit that includes a retry logic that is to retry access to a memory resource operatively coupled to the apparatus after receipt from the memory resource of a negative acknowledgment for an attempt to access the memory resource by the load/store unit; and

a congestion detection logic to output a signal that indicates that the memory resource is congested based on receipt from the memory resource of a consecutive number of negative acknowledgments in response to access requests to the memory resource.

2. (Previously Presented) The apparatus of claim 1 further comprising a congestion control logic to disable the retry logic from retry accesses to the memory resource based on receipt of the signal from the congestion detection logic.

3. (Previously Presented) The apparatus of claim 2, wherein the congestion control logic is to exponentially increase the delay after the congestion detection logic is to detect congestion while the memory resource is currently congested.

4. (Previously Presented) The apparatus of claim 2, wherein the congestion control logic is to exponentially decrease the delay after the congestion detection logic receive a number of positive acknowledgments in response to access requests to the memory resource.

5. (Previously Presented) A processor comprising:

a functional unit to attempt to access data from memory coupled to the processor based on an access request, wherein the functional unit is to retry attempts to access of the data based on other access requests after receipt from the memory of a negative acknowledgment in response to the attempt to access the data; and

a congestion detection logic to detect congestion of access of the data from the memory based on receipt from the memory of a consecutive number of negative acknowledgments that exceed a threshold prior to access of the data; and

a congestion control logic to disable the functional unit from the attempts to access the data from the memory for a time period after congestion is detected.

6. (Original) The processor of claim 5, wherein the congestion control logic is to exponentially increase the time period after the congestion detection logic is to detect congestion while access to other data in the memory is congested.

7. (Previously Presented) The processor of claim 6, wherein the congestion control logic is to exponentially decrease the time period after the congestion detection logic receives a number of positive acknowledgments in response to attempts to access data in the memory.

8. (Previously Presented) A processor comprising:

a functional unit to attempt to access a cache line in a cache memory coupled to the processor based on an access request, wherein the functional unit is to retry attempts to access the cache line based on additional access requests after receipt from the cache memory of a negative acknowledgment in response to the attempt to access the data;

a congestion detection logic to detect congestion of access of the cache line in the cache memory based on an average number of negative acknowledgments received from the cache memory that exceed a threshold prior to access of the data; and

a congestion control logic to disable the functional unit from attempts to access the cache line in the cache memory for a time period after congestion is detected.

9. (Previously Presented) The processor of claim 8, wherein the average number of negative acknowledgments is within a window and wherein the congestion detection logic is to move the window over time of attempts to access the cache line by the functional unit.

10. (Original) The processor of claim 8, wherein the congestion control logic is to exponentially increase the time period after the congestion detection logic is to detect congestion while access of other cache lines in the cache memory is congested.

11. (Previously Presented) The processor of claim 8, wherein the congestion control logic is to exponentially decrease the time period after the congestion detection logic receives a number of positive acknowledgments in response to attempts to access other cache lines in the cache memory.

12. (Previously Presented) A system comprising:

a cache memory to store data; and

a first processor to attempt to access data from the cache memory based on access requests, wherein the first processor includes a congestion detection logic to detect congestion of access to the data from the cache memory based on receipt from the cache memory of a consecutive number of negative acknowledgments in response to the access requests.

13. (Original) The system of claim 12 further comprising:

a second processor associated with the cache memory;

a hub controller to receive the access requests from the first processor, the hub controller to forward the access requests to the second processor, wherein the second processor is to determine whether the data in the cache memory is accessible.

14. (Previously Presented) The system of claim 13, wherein the second processor is to transmit a negative acknowledgment back to the first processor through the hub controller if the data is not accessible, the second processor to transmit a positive acknowledgment back to the first processor through the hub controller if the data is accessible.

15. (Original) The system of claim 12, wherein the first processor further comprises a congestion control logic to disable the first processor from transmitting the access requests if the congestion detection logic determines that access to the data is congested.

16. (Original) The system of claim 12, wherein the congestion control logic is to disable the first processor from transmitting the access requests for a time period, wherein the time period is based on an exponential back off delay operation.

17. (Previously Presented) A system comprising:

a memory resource; and

a first processor having a load/store functional unit, the load/store functional unit to attempt to access the memory resource based on access requests, wherein the first processor includes a congestion detection logic to detect congestion of access of the memory resource based on a consecutive number of negative acknowledgments received from the memory resource in response to the access requests prior to receipt received from the memory resource of a positive acknowledgment in response to one of the access requests within a first time period.

18. (Previously Presented) The system of claim 17 further comprising:

a second processor associated with the memory resource;

a hub controller to receive the access requests from the first processor, the hub controller to forward the access requests to the second processor, wherein the second processor is to determine whether the memory resource is accessible.

19. (Previously Presented) The system of claim 18, wherein the second processor is to transmit a negative acknowledgment back to the first processor through the hub controller if the memory resource is not accessible, the second processor to transmit a positive acknowledgment back to the first processor through the hub controller if the memory resource is accessible.

20. (Previously Presented) The system of claim 17, wherein the first processor further comprises a congestion control logic to disable the load/store functional unit from attempting to access the memory resource if the congestion detection logic is to detect congestion of access of the memory resource.

21. (Previously Presented) The system of claim 17, wherein the congestion control logic is to disable the load/store unit from attempts to access the memory resource for a second time period, wherein the second time period is based on an exponential back off delay operation.

22. (Previously Presented) A system comprising:

a cache memory to include a number of cache lines for storage of data; and

at least two processors, wherein a first processor of the at least two processors is to attempt to access the data in one of the number of cache lines in the cache memory based on access requests, wherein the first processor includes a congestion detection logic to detect congestion of access of a first cache line of the number of cache lines in the cache memory based on a ratio of a number of negative acknowledgments to a number of positive acknowledgments received from the cache memory in response to the access requests.

23. (Original) The system of claim 22, wherein a second processor of the at least two processors is associated with the cache memory and wherein the system further comprises a hub controller, the hub controller to receive the access requests from the first processor, the hub controller to forward the access requests to the second processor, wherein the second processor is to determine whether the one of the number of cache lines is accessible.

24. (Previously Presented) The system of claim 23, wherein the second processor is to transmit a negative acknowledgment back to the first processor through the hub controller if the one of the number of cache lines is not accessible, the second processor to transmit a positive acknowledgment back to the first processor through the hub controller if the one of the number of cache lines is accessible.

25. (Original) The system of claim 22, wherein the first processor further comprises a congestion control logic to disable, for a time period, the first processor to attempt to access the data if the congestion detection logic is to detect congestion of access of the first cache line.

26. (Original) The system of claim 25, wherein the congestion control logic is to exponentially increase the time period after the congestion detection logic is to detect congestion while access to other cache lines in the cache memory.

27. (Previously Presented) A method comprising:

transmitting access requests, by a first processor, to access data in a memory;

receiving, by the first processor, a positive acknowledgment or a negative acknowledgment from a second processor that is associated with the memory based on one of the number of access requests; and

detecting congestion of the data in the memory based on receipt, by the first processor, of a consecutive number of negative acknowledgments from the second processor that exceed a first threshold, prior to receipt, by the first processor, of a positive acknowledgment from the second processor.

28. (Previously Presented) The method of claim 27 further comprising controlling access to the data in the memory if the consecutive number of negative acknowledgments, received by the first processor, exceeds the first threshold, prior to receipt of the positive acknowledgment.

29. (Original) The method of claim 28, wherein controlling access to the data in the memory comprises disabling transmitting of the access requests, by the first processor, for a time period.

30. (Previously Presented) The method of claim 29, wherein controlling access to the data in the memory comprises exponentially increasing the time period upon determining that the congestion is detected for other data in the memory while the time period has not expired.

31. (Previously Presented) A method comprising:

accessing, by at least one processor, a memory resource based on an access request;

receiving a positive acknowledgment if the memory resource is accessible;

receiving a negative acknowledgment from the memory resource if the memory resource is not accessible;

retrying accessing, by the at least one processor, of the memory resource based on a number of access requests; and

detecting congestion of the memory resource based on receipt, by the at least one processor, from the memory resource of a consecutive number of negative acknowledgments that exceed a first threshold within a time period, prior to receiving a positive acknowledgment.

32. (Previously Presented) The method of claim 31 further comprising controlling access to the memory resource if the consecutive number of negative acknowledgment, received by the at least one processor, exceeds the first threshold, prior to receipt of the positive acknowledgment.

33. (Previously Presented) The method of claim 31, wherein controlling access to the memory resource comprises disabling transmitting of the access requests, by the first processor, for a time period.

34. (Previously Presented) A computer storage medium that provides instructions, which when executed by a machine, cause said machine to perform operations comprising:

transmitting access requests, by a first processor, to access data in a memory;

receiving, by the first processor, a positive acknowledgment or a negative acknowledgment from a second processor that is associated with the memory based on one of the number of access requests; and

detecting congestion of the data in the memory based on receipt, by the first processor, of a consecutive number of negative acknowledgments from the second processor that exceed a first threshold, prior to receipt, by the first processor, of a positive acknowledgment from the second processor.

35. (Previously Presented) The computer storage medium of claim 34 further comprising controlling access to the data in the memory if the consecutive number of negative acknowledgments, received by the first processor, exceeds the first threshold, prior to receipt of the positive acknowledgment.

36. (Previously Presented) The computer storage medium of claim 35, wherein controlling access to the data in the memory comprises disabling transmitting of the access requests, by the first processor, for a time period.

37. (Previously Presented) The computer storage medium of claim 36, wherein controlling access to the data in the memory comprises exponentially increasing the time period upon determining that the congestion is detected for other data in the memory while the time period has not expired.

38. (Previously Presented) A computer storage medium that provides instructions, which when executed by a machine, cause said machine to perform operations comprising:

accessing, by at least one processor, a memory resource based on an access request;

receiving a positive acknowledgment if the memory resource is accessible;

receiving a negative acknowledgment from the memory resource if the memory resource is not accessible;

retrying accessing, by the at least one processor, of the memory resource based on a number of access requests; and

detecting congestion of the memory resource based on receipt, by the at least one processor, from the memory resource of a consecutive number of negative acknowledgments that exceed a first threshold within a time period, prior to receiving a positive acknowledgments.

39. (Previously Presented) The computer storage medium of claim 38 further comprising controlling access to the memory resource if the consecutive number of negative acknowledgments, received by the at least one processor, exceeds the first threshold, prior to receipt of the positive acknowledgment.

40. (Previously Presented) The computer storage medium of claim 39, wherein controlling access to the memory resource comprises disabling transmitting of the access requests, by the first processor, for a time period.

41. (Previously Presented) The apparatus of claim 1, wherein the memory resource comprises nonvolatile memory.

42. (Previously Presented) The system of claim 17, wherein the memory resource comprises nonvolatile memory.

43. (Previously Presented) The method of claim 31, wherein the memory resource comprises nonvolatile memory.

44. (Previously Presented) The computer storage medium of claim 38, wherein the memory resource comprises nonvolatile memory.

45. (New) A system comprising:

a network;

a plurality of processors connected to the network, wherein the plurality processors includes a first processor; and

system memory, wherein the system memory includes remote memory that is across the network from the first processor, wherein the remote memory includes a cache memory to store data;

wherein the first processor is configured to attempt to access the data from the cache memory based on access requests and wherein the first processor includes a congestion detection logic to detect congestion of access to the data from the cache memory based on receipt from the cache memory of a consecutive number of negative acknowledgments in response to the access requests.

46. (New) The system of claim 45, wherein the access requests comprise retry attempts by the first processor to access the data from the cache memory.